IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
Before the Board of Patent Appeals and Interferences

In re Patent Application of: NIGHTINGALE et al
Serial No. 10/079,811
Filed: February 22, 2002
Title: SOFTWARE AND HARDWARE SIMULATION

Atty Dkt. 550-318
C# M#
TC/A.U.: 2128
Examiner: S. Patel
Date: September 20, 2006

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

☐ **Correspondence Address Indication Form Attached.**

☐ **NOTICE OF APPEAL**
Applicant hereby **appeals** to the Board of Patent Appeals and Interferences from the last decision of the Examiner twice/finally rejecting applicant's claim(s). $500.00 (1401)/$250.00 (2401) $

☒ An appeal **BRIEF** is attached in the pending appeal of the above-identified application $500.00 (1402)/$250.00 (2402) $ 500.00

☐ Credit for fees paid in prior appeal without decision on merits -$ ( )

☐ A reply brief is attached. **(no fee)**

☒ Petition is hereby made to extend the current due date so as to cover the filing date of this paper and attachment(s)
One Month Extension $120.00 (1251)/$60.00 (2251)
Two Month Extensions $450.00 (1252)/$225.00 (2252)
Three Month Extensions $1020.00 (1253/$510.00 (2253)
Four Month Extensions $1590.00 (1254/$795.00 (2254) $ 120.00

☐ "Small entity" statement attached.

Less month extension previously paid on -$( )

TOTAL FEE ENCLOSED $ 620.00

Any future submission requiring an extension of time is hereby stated to include a petition for such time extension. The Commissioner is hereby authorized to charge any deficiency, or credit any overpayment, in the fee(s) filed, or asserted to be filed, or which should have been filed herewith (or with any paper hereafter filed in this application by this firm) to our **Account No. 14-1140.** A duplicate copy of this sheet is attached.

901 North Glebe Road, 11th Floor
Arlington, Virginia 22203-1808
Telephone: (703) 816-4000
Facsimile: (703) 816-4100
SCS:kmm

NIXON & VANDERHYE P.C.
By Atty: Stanley C. Spooner, Reg. No. 27,393

09/21/2006 MAHMED1 00000093 10079811
01 FC:1251 120.00 OP

Signature:

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Patent Application of	Confirmation No.: 5550
NIGHTINGALE et al	Atty. Ref.: 550-318
Serial No. 10/079,811	Group: 2128
Filed: February 22, 2002	Examiner: S. Patel

For: SOFTWARE AND HARDWARE SIMULATION

* *

APPEAL BRIEF

On Appeal From Group Art Unit 2128

Stanley C. Spooner
NIXON & VANDERHYE P.C.
11th Floor, 901 North Glebe Road
Arlington, Virginia 22203
(703) 816-4028
Attorney for Appellant

TABLE OF CONTENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Patent Application of

NIGHTINGALE et al — Atty. Ref.: 550-318

Serial No. 10/079,811 — Group: 2128

Filed: February 22, 2002 — Examiner: S. Patel

For: SOFTWARE AND HARDWARE SIMULATION

* *

September 20, 2006

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

Sir:

I. REAL PARTY IN INTEREST

The real party in interest in the above-identified appeal is ARM Limited by virtue of an assignment of rights from the inventors to ARM Limited recorded April 8, 2002 at Reel 12773, Frame 6.

II. RELATED APPEALS AND INTERFERENCES

There are believed to be no related appeals, interferences or judicial proceedings with respect to the present application and appeal, save for Appellants' previously filed Pre-Appeal Brief Request for Review.

III. STATUS OF CLAIMS

Claims 1-16 stand rejected in the outstanding Final Rejection as being obvious under 35 USC §103 over Hollander (U.S. Patent 6,182,258) in view of Platt (U.S. Patent 5,835,764), with claims 3-5 rejected under 35 USC §103 as unpatentable over the Hollander/Platt combination in further view of Campbell (U.S. Patent 6,408,009) and claim 13 rejected under 35 USC §103 as unpatentable over the Hollander/Platt combination in further view of Harmon (U.S. Patent 6,810,373). The rejections of claims 1-16 are appealed.

IV. STATUS OF AMENDMENTS

No further response has been submitted with respect to the Final Official Action in this application other than the filing of a Pre-Appeal Brief Request for Review which decision was mailed July 13, 2006.

V. SUMMARY OF THE CLAIMED SUBJECT MATTER

Appellants' specification and figures provide an explanation of the claimed invention set out in independent claims 1, 15 and 16, with each claimed structure and method step addressed as to its location in the specification and in the figures.

"1. A method of simulating a system having a software component [2 as disclosed in Figure 1 and discussed on page 8, line 20 to page 9, line 18 and

elsewhere in the specification] and a hardware component [4 as disclosed in Figure 1 and discussed on page 8, line 20 to page 9, line 18 and elsewhere in the specification], said method comprising the steps of:

(i) modelling operation of said software component using a software simulator [item 16 as disclosed in Figure 2 and discussed on page 9, line 20 to page 10, line 16, and item 30 as disclosed in Figure 4 and discussed on page 11, line 27 to page 12, line 12 and elsewhere in the specification];

(ii) modelling operation of said hardware component using a hardware simulator [item 10 as disclosed in Figure 2 and discussed on page 9, line 20 to page 10, line 16, and item 28 as disclosed in Figure 4 and discussed on page 11, line 27 to page 12, line 12 and elsewhere in the specification];

(iii) linking [items 12 and 14 as disclosed in Figure 2 and discussed on page 9, line 20 to page 10, line 16, and item 32 as disclosed in Figure 4 and discussed on page 11, line 27 to page 12, line 12 and elsewhere in the specification] said hardware simulator [10, 28] and said software simulator [16, 30] to model interaction between said modelled operation of said hardware component and said modelled operation of said software component;

(iv) generating with a test controller [item 18 as disclosed in Figure 2 and discussed on page 10, lines 1-16, and item 26 as disclosed in Figure 4 and discussed on page 11, line 27 to page 12, line 12 and elsewhere in the specification], during said modelling of software and hardware components and

said interaction, a software stimulus [item 22 disclosed in Figure 2 and discussed on page 10, lines 1-16 and elsewhere in the specification] for said software component and a hardware stimulus [item 20 disclosed in Figure 2 and discussed on page 10, lines 1-16 and elsewhere in the specification] for said hardware component, said software stimulus and said hardware stimulus are associated so as to permit verification of correct interoperability of said software component and said hardware component, wherein said modelled interaction between said software component and said hardware component proceeds independently of said test controller;

(v) modelling [using items 16 or 30 – see above] the response of said software component to said software stimulus; and

(vi) modelling [using items 10 or 28 – see above] the response of said hardware component to said hardware stimulus, wherein said software stimulus is passed to said software simulator by issuing a remote procedure call from said test controller to said software simulator [as discussed on page 12, lines 8-12]."

"15. Apparatus for simulating a system having a software component [2 as disclosed in Figure 1 and discussed on page 8, line 20 to page 9, line 18 and elsewhere in the specification] and a hardware component [4 as disclosed in Figure 1 and discussed on page 8, line 20 to page 9, line 18 and elsewhere in the specification], said apparatus comprising:

(i) a software simulator [item 16 as disclosed in Figure 2 and discussed on page 9, line 20 to page 10, line 16, and item 30 as disclosed in Figure 4 and discussed on page 11, line 27 to page 12, line 12 and elsewhere in the specification] for modelling operation of said software component;

(ii) a hardware simulator [item 10 as disclosed in Figure 2 and discussed on page 9, line 20 to page 10, line 16, and item 28 as disclosed in Figure 4 and discussed on page 11, line 27 to page 12, line 12 and elsewhere in the specification] for modelling operation of said hardware component;

(iii) means for linking [items 12 and 14 as disclosed in Figure 2 and discussed on page 9, line 20 to page 10, line 16, and item 32 as disclosed in Figure 4 and discussed on page 11, line 27 to page 12, line 12 and elsewhere in the specification] said hardware simulator [10, 28] and said software simulator [16,30] to model interaction between said modelled operation of said hardware component and said modelled operation of said software component;

(iv) a test controller [item 18 as disclosed in Figure 2 and discussed on page 10, lines 1-16, and item 26 as disclosed in Figure 4 and discussed on page 11, line 27 to page 12, line 12 and elsewhere in the specification] for generating, during modelling of said software and hardware components and said interaction, a software stimulus [item 22 disclosed in figure 2 and discussed on page 10, lines 1-16 and elsewhere in the specification] for said software component and a hardware stimulus [item 20 disclosed in figure 2 and discussed on page 10, lines

1-16 and elsewhere in the specification] for said hardware component, said software stimulus and said hardware stimulus are associated so as to permit verification of correct interoperability of said software component and said hardware component, wherein said modelled interaction between said software component and said hardware component proceeds independently of said test controller;

(v) said software simulator includes means for modelling [items 16 or 30 – see above] a response of said software component to said software stimulus; and

(vi) said hardware simulator includes means for modelling [items 10 or 28 – see above] a response of said hardware component to said hardware stimulus, wherein said software stimulus is passed to said software simulator by issuing a remote procedure call from said test controller to said software simulator [as discussed on page 12, lines 8-12]."

"16. A computer program product comprising a computer-readable medium [items 204, 206 and 210 shown in Figure 9 and discussed on page 16, lines 6-26 and elsewhere in the specification] for controlling a computer [200 shown in Figure 9 and discussed on page 16, lines 6-26 and elsewhere in the specification] to simulate a system having a software component and a hardware component, said computer program product comprising:

(i) software simulator logic for modelling operation of said software component [item 16 as disclosed in Figure 2 and discussed on page 9, line 20 to page 10, line 16, and item 30 as disclosed in Figure 4 and discussed on page 11, line 27 to page 12, line 12 and elsewhere in the specification];

(ii) hardware simulator logic for modelling operation of said hardware component [item 10 as disclosed in Figure 2 and discussed on page 9, line 20 to page 10, line 16, and item 28 as disclosed in Figure 4 and discussed on page 11, line 27 to page 12, line 12 and elsewhere in the specification];

(iii) logic linking said hardware simulator logic [10, 28] and said software simulator logic [16, 30] to model interaction between said operation of said hardware component and said modelled operation of said software component [items 12 and 14 as disclosed in Figure 2 and discussed on page 9, line 20 to page 10, line 16, and item 32 as disclosed in Figure 4 and discussed on page 11, line 27 to page 12, line 12 and elsewhere in the specification];

(iv) test controller logic [item 18 as disclosed in Figure 2 and discussed on page 10, lines 1-16, and item 26 as disclosed in Figure 4 and discussed on page 11, line 27 to page 12, line 12 and elsewhere in the specification] for generating a software stimulus [item 22 disclosed in figure 2 and discussed on page 10, lines 1-16 and elsewhere in the specification] for said software component and a hardware stimulus [item 20 disclosed in figure 2 and discussed on page 10, lines 1-16 and elsewhere in the specification] for said hardware component, said generating

occurring during modelling of said software component and said hardware component and said modelled interaction, said software stimulus and said hardware stimulus being associated so as to permit verification of correct interoperability of said software component and said hardware component, wherein said interaction between said software component and said hardware component proceeds independently of said test controller logic;

(v) said software simulator logic includes logic for modelling a response of said software component to said software stimulus [using items 16 or 30 – see above]; and

(vi) said hardware simulator logic includes logic for modelling a response of said hardware component to said hardware stimulus [using items 10 or 28 – see above], wherein said software stimulus is passed to said software simulator logic by issuing a remote procedure call from said test controller logic to said software simulator logic [as discussed on page 12, lines 8-12]."

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1, 2, 6, 7-12, 14, 15 and 16 stand rejected under 35 USC §103 as unpatentable over Hollander (U.S. Patent 6,182,258) in view of Platt (U.S. Patent 5,835,764).

Claims 3-5 stand rejected under 35 USC §103 as being unpatentable over Hollander and Platt in further view of Campbell (U.S. Patent 6,408,009).

Claim 13 stands rejected under 35 USC §103 as unpatentable over Hollander and Platt in further view of Harmon (U.S. Patent 6,810,373).

VII. ARGUMENT

Appellants' arguments include the fact that the burden is on the Examiner to first and foremost properly construe the language of the claims to determine what structure and/or method steps are covered by that claim. After proper construction of the claim language, the burden is also on the Examiner to demonstrate where among a plurality of references (in the case of an obviousness rejection) there is a teaching of each of the structures and/or method steps (and any claimed interrelationship) recited in independent claims 1, 15 and 16. The Court of Appeals for the Federal Circuit has stated with respect to rejections under 35 USC §103 in *In re Rouffet*, 47 USPQ2d 1453, 1458 (Fed. Cir. 1998)

> "to prevent the use of hindsight based on the invention to defeat patentability of the invention, this court **requires** the examiner to show a **motivation** to combine the references that create the case of obviousness. In other words, the Examiner **must show reasons** that the skilled artisan, confronted with the same problems as the inventor and with no knowledge of the claimed invention, would select the elements from the cited prior art references for combination in the manner claimed." (emphasis added).

With the exception of independent claims 15 and 16, all claims depend from method claim 1. Thus, in section A, Appellants will focus on the failure of the Examiner to establish how or where the recited method steps set out in claim 1

(and the corresponding structures in Claim 15 and logic in Claim 16) are disclosed in any of the cited and applied prior art references. Section B will confirm that the Examiner has not alleged that the Claim 1 method steps (and the corresponding structures in Claim 15 and logic in Claim 16) missing from Hollander are shown in any of the secondary Platt, Campbell or Harmon references and thus even if all cited references are combined, there is still no disclosure of the missing method steps, elements and logic. Section C points out that the Examiner has failed to provide any basis for combining the references as required by *In re Rouffet.*

A. The Examiner fails to point out how or where Hollander teaches all elements of Appellants' claimed invention

1. The Examiner fails to establish how or where Hollander teaches Appellants' claimed "(i) modelling" operation of a software component using a software simulator

Appellants' independent claim 1 requires "modelling operation of said software component using a software simulator" (with Claim 15 requiring a "software simulator" and Claim 16 requiring "software simulator logic"). The portion of the Final Rejection entitled "Response to Arguments" completely ignores this claimed requirement of claims 1, 15 and 16, perhaps indicating that the Examiner has no response to this argument.

The restatement of the rejection under 35 USC §103 on page 6 of the final rejection (as noted in the previous official action) does allege that Hollander is

directed to a method of simulating a system and in item "a". suggests that it teaches the claimed "modeling." The Examiner supports his contention by refering to column 8, lines 39-44 and column 10, lines 51-58 in Hollander seeming to suggest this contains some teaching or suggestion of the missing step, element or logic.

However, neither of the cited portions of the Hollander reference have anything to do with "modelling operation of said software component using a software simulator." The column 8 discussion relates to "a report generator module 24 [which] provides textual and graphical information on the test results" and the column 10 discussion deals with a "co-verification extension module 174." Since neither of these structures have anything to do with Appellants' claimed method step or corresponding structure or logic (in claims 15 and 16), Hollander simply fails to disclose claimed subject matter and therefore there can be no *prima facie* case of obviousness.

In fact, a more pertinent portion of the Hollander reference is believed to be at column 10, lines 24-28, where it is stated that "the user first creates a cycle-accurate model of the hardware apparatus 172 on which the external software program 168 is to be run." However, this clearly indicates that the software component is **not modeled** and instead **a real software program is simply run**, albeit on a model of the hardware apparatus.

Because Hollander contains no teaching of the claim 1 "modeling" step, the claim 15 "software simulator" and the claim 16 "software simulator logic",

Hollander does not support a rejection of claims 1, 15 and 16 under 35 USC §103 or any claims dependent thereon.

2. The Examiner fails to establish how or where Hollander teaches Appellants' claimed "(iv) generating" step wherein "said modelled interaction between said software component and said hardware component proceeds independently of said test controller"

In section (iv) of claim 1, relating to "generating" (in Claim 15 the "test controller" and in claim 16 the "test controller logic"), it is specified that "wherein said modelled interaction between said software component and said hardware component proceeds independently of said test controller."

The Examiner purports to address this claim requirement in the first paragraph on page 4 of the Final Rejection "Response to Arguments." The Examiner alleges that Hollander discloses the claimed method step and corresponding structures and logic at column 5, lines 21-24 and column 8, lines 24-32. Again the cited portion of Hollander has nothing to do with the claimed method step, element or logic.

The column 5 citation merely suggests that the Hollander invention "can perform any combination of static and dynamic checks" and that "the test generator module and the checker can constantly synchronize." While this language may be considered broad enough to cover Appellant's claimed subject matter, it does not disclose the claimed subject matter. It is noted that the issue is

that the test of obviousness is whether a prior art reference contains a disclosure of a claimed element, and not whether a prior art patent has broad claim coverage.

The second citation merely identifies that Hollander uses "a checker module 30" and the checker module along with the corresponding generate module 26 are elements that "can constantly synchronize." A brief reference to Hollander's Figure 1 will indicate that the test generator module 26 and the checker module 30 are portions of the test controller 22 (unlabeled in Figure 1).

Because the checker and the test generator are part of the test controller, it is impossible for the modelled interaction "between said software component and said hardware component [to proceed] **independently** of said test controller" as set out in Appellant's claims (emphasis added). Therefore this claimed interrelationship is clearly missing from the Hollander reference. In fact, Hollander suggests that the only interaction between software and hardware in the Hollander reference is that which is foreseen to occur at "pre-designated points" in the software (see Hollander at column 10, lines 43-45).

Thus, Hollander cannot discover the unexpected, which is the whole point of the present invention. Appellant's claimed interrelationship which proceeds "independently of said test controller" is a crucial feature of the present invention and allows a more realistic simulation of the interplay between software and hardware to be performed than was previously possible in Hollander and other prior art teachings.

Again, the subject matter of the independent claim "generating" step (and the corresponding "test controller" of claim 15 and the "test controller logic" of claim 16) and specifically claimed modelled interaction which proceeds "independently of said test controller" (present in all three independent claims) is not disclosed in the Hollander reference. Therefore the combination of this and other steps in the method claims, in the apparatus claims and similar logic in the logic claims cannot be obvious in view of the Hollander reference and therefore any further rejection of claims 1, 15 and 16 (and any claims dependent thereon) is respectfully traversed.

3. The Examiner fails to establish how or where Hollander teaches Appellants' claimed "(vi) modelling" the response of the hardware component wherein software stimulus "is passed to said software simulator by issuing a remote procedure call from said test controller to said software simulator"

The Examiner, in apparent admission that the Hollander reference has no disclosure relating to the use of a remote procedure call, cites the Platt reference as teaching Appellants' last "modelling" step in which the software stimulus is "passed to said software simulator by issuing **a remote procedure call** from said test controller to said software simulator." (emphasis added).

While Platt does suggest use of a procedure call, the Examiner ignores the problem that would be faced in applying the Platt solution to the Hollander system. The Examiner does not provide any indication of how one of ordinary

skill in the art would be directed to apply remote procedural calls as a communication means in the Hollander reference, in view of the fact that the test controller 166 and the external software 163 (as shown in Figure 5 of Hollander) **exist in entirely separate domains with no apparent communication link available** in order to implement the "remote procedure call" solution disclosed in Appellants' invention.

The Examiner's summary in the "Response to Arguments" portion of the Final Rejection on page 4, second paragraph, merely confirms that remote procedural calls are feasible in both the present invention and in the Platt reference. However, he ignores the well-known problem of how to incorporate such a procedure call disclosed in the Platt reference in the Hollander reference itself.

In view of the above, it would be clear to one of ordinary skill in the art that this could not be done, as there is no obvious communication link between the test controller and the external software in order to be able to incorporate the Platt "remote procedure call" procedure.

Thus, again there is no last "modelling" step which includes passing information by way of "a remote procedure call" (as set forth in Appellants' independent claims) disclosed in the Hollander/Platt combination. In fact, due to the Hollander problems noted above, Platt could not be incorporated into the Hollander solution.

B. The secondary references of Platt, Campbell and Harmon fail to teach or suggest the elements missing from Hollander

In the paragraph bridging pages 6 and 7 of the Final Rejection, the Examiner admits that the Hollander reference fails to disclose a method where the software stimulus is passed to a software simulator using a remote procedure call. While the Examiner alleges that Platt teaches this missing feature, there is no allegation by the Examiner anywhere in the Final Rejection that Platt or Campbell or Harmon contain any disclosure of Appellants' claimed "(i) modelling" operation, i.e., modelling operation of said software component **using a software simulator**.

Clearly, the Examiner has not only ignored the claimed "modelling operation of said software component" step in Appellants' independent claim 1 (and corresponding independent claims 15 and 16), but he has not even alleged that this featur is disclosed or somehow rendered obvious in any of the other cited secondary references. It appears that the Examiner was well aware of this omission because the "Response to Arguments" portion of the Final Rejection glaringly ignores this claimed requirement of claims 1, 15 and 16.

Similarly, the Examiner makes no allegation that Platt, Campbell or Harmon contain any disclosure or suggestion of Appellants' claimed "generating" step using a test controller "wherein said modelled interaction between said software component and said hardware component proceeds **independently of**

said test controller." In the discussion of the Platt reference in the first full paragraph of page 7 of the Final Rejection, the Examiner fails to address the claimed "generating" step (or the test controller of claim 15 or the test controller logic of claim 16). In the discussion of the Campbell reference on page 10 of the Final Rejection, the Examiner does not allege any disclosure of the "generating" step, and similarly in the paragraph discussing the Harmon reference on page 11, there is no allegation that there is any disclosure of the generating step.

In view of the above, not only does Hollander fail to disclose Appellants' claimed method step (and corresponding structure and logic in independent claims 15 and 16), there is no disclosure of these claim limitations in the secondary references. Therefore, even if all prior art were to be combined, there would be no basis for a rejection of claims 1, 15 and 16 or any claims dependent on claim 1. Thus, the rejection of claims 1, 2, 6, 7-12, 14, 15 and 16 over Hollander/Platt, the rejection of claims 3-5 under §103 over Hollander/Platt in view of Campbell and the rejection of claim 13 under §103 over the Hollander/Platt combination in view of Harmon are completely unsupported by the outstanding Final Rejection.

As the Court of Appeals for the Federal Circuit held in the case of *In re Fine*, 5 USPQ2d 1596, 1598 (Fed. Cir. 1988), "the PTO has the burden under §103 to establish a *prima facie* case of obviousness." The Court also held that the PTO "can satisfy this burden only by showing some objective teaching in the prior art . . . ."

The Examiner has failed to demonstrate any teaching of at least three claimed method steps or interrelationships between method steps as recited in claim 1 (and corresponding apparatus and logic in claims 15 and 16). Thus, even if all four references were combined, they would not disclose the limitations discussed above in independent claims 1, 15 and 16 or claims dependent from claim 1.

C. The Examiner fails to establish any reason or motivation for combining references

As noted above, the Court of Appeals for the Federal Circuit has consistently held that the burden is on the Examiner to establish some "motivation" to combine the references. In explanation, the Court stated that the Examiner "must show reasons" that one of ordinary skill in the art confronted with the same problems as the inventor, but "with no knowledge of the claimed invention," would select the elements from the prior art.

The Examiner has simply ignored this important requirement which was imposed by the Court of Appeals for the Federal Circuit in order to prevent "the use of hindsight based on the invention to defeat patentability of the invention."

With respect to the basis for combining Hollander and Platt, the Examiner offers the conclusory statement that "it would have been obvious to one of ordinary skill in the art to combine the teachings of Hollander and Platt." The Examiner then suggests that a motivation for doing so would have been to develop

a method of communication that exhibits high performance, citing Platt at column 10, lines 16-22. "High performance" is a term applied to almost every computer system in the world.

Moreover, the cited portion of the Platt reference merely says "it has been found that a transactional system in accordance with the present invention allows full operating system functionality whilst at the same time providing high performance transaction processing." This provides no "reason" or "motivation" for combining portions of appellants claims which are allegedly disclosed in the Hollander reference with the portions allegedly disclosed in the Platt reference.

The Examiner has provided no underlying support as to the Hollander and Platt patents evidencing artisans confronted with the same problems as in the claimed invention. The Examiner has failed to identify, other than the desire for "high performance," method steps or elements which would supply a solution to the problems. There is simply no support for the combination of Hollander and Platt.

Because the Examiner has failed to meet his duty of establishing some "reason" or "motivation" for combining bits and pieces of the Hollander and Platt references, he has failed to support the rejection of claims 1, 2, 6, 7-12, 14, 15 and 16 under 35 USC §103 and any further rejection thereunder is respectfully traversed.

To the extent that the Hollander/Platt combination is used in the rejections of claims 3-5 (Hollander/Platt/Campbell) and claim 13 (Hollander/Platt/Harmon), the same lack of support for the Hollander/Platt combination is also evident and thus those rejections fail.

In order to combine the Hollander/Platt references with Campbell, the Examiner again is required to show some "reason" or "motivation" for that combination. On page 10 of the Final Rejection, the Examiner makes the obligatory conclusory statement that "it would have been obvious to combine the teachings of Hollander, Platt and Campbell."

The sole allegation as to a motivation for combining bits and pieces of these three references is the Examiner's conclusion that it "would be to incorporate an effective method for determining that the stimulus has been received and therefore the simulation can begin." Again, the Examiner's use of "effective method" is simply an indication that he can find no "reason" or "motivation" in any of the three cited references. There is no indication that any of the three references relate to the problems solved in the present invention. There is no indication that the structures or method steps disclosed in the various references could be combined in the manner of Appellants' independent claims 1, 15 and 16, save for Appellants' specification and drawings. The Examiner simply fails to support his rejection of claims 3-5 based upon 35 USC §103 over

Hollander/Platt/Campbell and any further rejection thereunder is respectfully traversed.

The Examiner rejects claim 13 under 35 USC §103 as unpatentable over the Hollander/Platt combination in further view of Harmon. As set out on page 11 of the Final Rejection, the Examiner again concludes that "it would have been obvious to combine the teachings of Harmon with the above combination of Hollander and Platt." However, the sole motivation for doing so, according to the Examiner, is "to increase the performance of the system."

Interestingly, the Examiner cites Platt as support for his conclusion as to "motivation" and specifically identifies column 2, lines 66-67 and column 3, lines 1, 2 and 46-57) as purportedly providing the required motivation. However, Platt at column 2, lines 66 and 67 merely states "[a] sync point is taken at the end of every transaction. Whenever a sync point is taken, the system goes through a commit protocol to ensure consistency of data. An application can explicitly request a sync point to be taken at any . . ." (Platt, column 2, line 67 through column 3, line 2).

Clearly, the above quote from Platt has absolutely nothing to do with the Examiner's contention that use of an ISS (instruction set simulator) increases the speed of verification or that such verification speed is "greatly increased." The remaining citation to the Platt reference at column 3, lines 46-57 contains a discussion of the CCM (concurrency control manager) which is "the heart of the

transactional system." It then goes on to discuss the concurrency control protocol, but nowhere is there any indication that this relates to an instruction set simulator or that by use of an instruction set simulator any speed of verification is "greatly increased." In fact, there is believed to be no disclosure of anything relating to speed of verification, let alone any increase of speed of verification.

Again, the Examiner has simply ignored the Court's requirement that "the examiner [to] show a motivation to combine the references that create the case of obviousness." It seems obvious that the Examiner has merely alleged a "motivation" with respect to the combinations of these references and has cited portions of the references in hopes that the Board will not carefully examine the matter. A proper examination of the cited portions of the prior art will establish that they do not support the Examiner's contentions.

The Examiner has again failed to provide any motivation for combining the references. Absent a motivation for combining references, the rejection of claim 13 over the Hollander/Platt/Harmon reference falls.

VIII. CONCLUSION

As noted above, the Examiner's Final Rejection ignores the lack of the claimed "modelling operation" set out in item (i) in Appellants' independent claims. This failure alone removes any *prima facie* basis for an obviousness rejection. As noted above, the Final Rejection also does not point to any teaching

in the Hollander reference wherein modelled interaction between software components and hardware components proceeds "independently of said test controller." In fact, the cited portion of Hollander has nothing to do with this claimed step, structure or logic. The lack of this feature as well in Hollander clearly avoids any obviousness rejection under 35 USC §103. Finally, the lack of the final "modelling" aspect of Appellants' invention in which communication from said test controller to said software simulator proceeds by way of a "remote procedure call," while disclosed in the Platt reference, is not necessarily combinable with Hollander because there is no communication link available.

The Examiner does not dispute that the elements missing from Hollander are also missing from the Platt, Campbell and Harmon references. Thus even if combined, all four prior art references would not render obvious the claim invention. Moreover, there is no indication as to why one of ordinary skill in the art would think to combine these dissimilar references in the claimed fashion, since there is no means for communication disclosed in the Hollander reference.

As a result of the above, there is simply no support for the rejections of Appellants' independent claims 1, 15 and 16 or claims dependent thereon under the provisions of 35 USC §103. Thus, and in view of the above, the rejection of claims 1-16 under 35 USC §103 is clearly in error and reversal thereof by this Honorable Board is respectfully requested.

Respectfully submitted,

NIXON & VANDERHYE P.C.

By: _____

Stanley C. Spooner
Reg. No. 27,393

SCS:kmm
Enclosure

IX. CLAIMS APPENDIX

1. A method of simulating a system having a software component and a hardware component, said method comprising the steps of:

(i) modelling operation of said software component using a software simulator;

(ii) modelling operation of said hardware component using a hardware simulator;

(iii) linking said hardware simulator and said software simulator to model interaction between said modelled operation of said hardware component and said modelled operation of said software component;

(iv) generating with a test controller, during said modelling of software and hardware components and said interaction, a software stimulus for said software component and a hardware stimulus for said hardware component, said software stimulus and said hardware stimulus are associated so as to permit verification of correct interoperability of said software component and said hardware component, wherein said modelled interaction between said software component and said hardware component proceeds independently of said test controller;

(v) modelling the response of said software component to said software stimulus; and

(vi) modelling the response of said hardware component to said hardware stimulus, wherein said software stimulus is passed to said software simulator by issuing a remote procedure call from said test controller to said software simulator.

2. A method as claimed in claim 1, wherein said test controller issues said remote procedure call by writing call data specifying said software stimulus to a shared memory, said software simulator reading call data from said shared memory to trigger modelling of operation of said software component in response to said software stimulus.

3. A method as claimed in claim 2, wherein said test controller sets a start flag within said shared memory to indicate to said software simulator that said shared memory contains call data specifying a software stimulus to be modelled.

4. A method as claimed in claim 3, wherein said software simulator polls said start flag to determine if there is a software stimulus to be modelled.

5. A method as claimed in claim 3, wherein said software simulator resets said start flag to indicate to said test controller that modelling of said software stimulus has been completed.

6. A method as claimed in claim 2, wherein said call data includes one or more of:

(i) data identifying a software routine to be modelled within said software component; and

(ii) variable data to be used in responding to said software stimulus.

7. A method as claimed in claim 1, wherein said hardware component is a hardware peripheral within a data processing system.

8. A method as claimed in claim 1, wherein said software component is a software driver for said hardware component.

9. A method as claimed in claim 1, further comprising monitoring modelled signals at an interface with said hardware component that are generated in response to simulation of said software component and said hardware component.

10. A method as claimed in claim 9, wherein said modelled signals are monitored for compliance with rules defining permitted values for said modelled signals.

11. A method as claimed in claim 1, wherein said software simulator is monitored to determine coverage of a range of software stimuli that may be applied to said software simulator.

12. A method as claimed in claim 1, wherein said hardware simulator is monitored to determine coverage of a range of hardware stimuli that may be applied to said hardware simulator.

13. A method as claimed in claim 1, wherein said software simulator is an instruction set simulator that serves to model execution of software program instruction by a data processing core.

14. A method as claimed in claim 1, further comprising monitoring said hardware simulator to detect expected changes of state within said hardware component occurring in response to said software stimulus.

15. Apparatus for simulating a system having a software component and a hardware component, said apparatus comprising:

(i) a software simulator for modelling operation of said software component;

(ii) a hardware simulator for modelling operation of said hardware component;

(iii) means for linking said hardware simulator and said software simulator to model interaction between said modelled operation of said hardware component and said modelled operation of said software component;

(iv) a test controller for generating, during modelling of said software and hardware components and said interaction, a software stimulus for said software component and a hardware stimulus for said hardware component, said software stimulus and said hardware stimulus are associated so as to permit verification of correct interoperability of said software component and said hardware component, wherein said modelled interaction between said software component and said hardware component proceeds independently of said test controller;

(v) said software simulator includes means for modelling a response of said software component to said software stimulus; and

(vi) said hardware simulator includes means for modelling a response of said hardware component to said hardware stimulus, wherein said software stimulus is passed to said software simulator by issuing a remote procedure call from said test controller to said software simulator.

16. A computer program product comprising a computer-readable medium for controlling a computer to simulate a system having a software component and a hardware component, said computer program product comprising:

(i) software simulator logic for modelling operation of said software component;

(ii) hardware simulator logic for modelling operation of said hardware component;

(iii) logic linking said hardware simulator logic and said software simulator logic to model interaction between said operation of said hardware component and saiod modelled operation of said software component;

(iv) test controller logic for generating a software stimulus for said software component and a hardware stimulus for said hardware component, said generating occurring during modelling of said software component and said hardware component and said modelled interaction, said software stimulus and said hardware stimulus being associated so as to permit verification of correct interoperability of said software component and said hardware component, wherein said interaction between said software component and said hardware component proceeds independently of said test controller logic;

(v) said software simulator logic includes logic for modelling a response of said software component to said software stimulus; and

(vi) said hardware simulator logic includes logic for modelling a response of said hardware component to said hardware stimulus, wherein said software stimulus is passed to said software simulator logic by issuing a remote procedure call from said test controller logic to said software simulator logic.

X. EVIDENCE APPENDIX

None.

XI. RELATED PROCEEDINGS APPENDIX

None.